

Foreman - Bug #296

Incorrect family_id references breaks migration

06/02/2010 01:22 AM - Frank Sweetser

Status:	Closed	
Priority:	Normal	
Assignee:	Ohad Levy	
Category:	DB migrations	
Target version:	0.1-5	
Difficulty:		Fixed in Releases:
Triaged:		Found in Releases:
Bugzilla link:		Red Hat JIRA:
Pull request:		

Description

Upgrading to the latest 1.5 RC, and ran into this error when running db:migrate:

```
[root@jms foreman]# rake db:migrate --trace
(in /usr/share/foreman)
```

- Invoke db:migrate (first_time)
 - Invoke environment (first_time)
 - Execute environment
 - Execute db:migrate
- ```
CreateMediasOperatingSystemsAndMigrateData: migrating =====
rake aborted!
An error has occurred, all later migrations canceled:
```

missing attribute: family\_id

```
/usr/share/foreman/app/models/operatingsystem.rb:20:in `after_initialize'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/callbacks.rb:347:in `send'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/callbacks.rb:347:in `callback'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/base.rb:1657:in `send'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/base.rb:1657:in `instantiate_without_polymorphic_checks'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/base.rb:52:in `instantiate'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/base.rb:661:in `find_by_sql'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/base.rb:661:in `collect!'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/base.rb:661:in `find_by_sql'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/base.rb:1548:in `find_every'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/base.rb:1505:in `find_initial'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/base.rb:692:in `exists?'
./db/migrate//20100523141204_create_medias_operatingsystems_and_migrate_data.rb:7:in `up_without_benchmarks'
./db/migrate//20100523141204_create_medias_operatingsystems_and_migrate_data.rb:5:in `each'
./db/migrate//20100523141204_create_medias_operatingsystems_and_migrate_data.rb:5:in `up_without_benchmarks'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/migration.rb:282:in `send'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/migration.rb:282:in `migrate'
/usr/lib/ruby/1.8/benchmark.rb:293:in `measure'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/migration.rb:282:in `migrate'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/migration.rb:365:in `__send__'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/migration.rb:365:in `migrate'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/migration.rb:486:in `migrate'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/migration.rb:562:in `call'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/migration.rb:562:in `ddl_transaction'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/migration.rb:485:in `migrate'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/migration.rb:472:in `each'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/migration.rb:472:in `migrate'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/migration.rb:400:in `up'
/usr/share/foreman/vendor/rails/activerecord/lib/active_record/migration.rb:383:in `migrate'
/usr/share/foreman/vendor/rails/railties/lib/tasks/databases.rake:116
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:635:in `call'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:635:in `execute'
```

```
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:630:in `each'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:630:in `execute'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:596:in `invoke_with_call_chain'
/usr/lib/ruby/1.8/monitor.rb:238:in `synchronize'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:589:in `invoke_with_call_chain'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:582:in `invoke'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:2050:in `invoke_task'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:2028:in `top_level'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:2028:in `each'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:2028:in `top_level'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:2067:in `standard_exception_handling'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:2022:in `top_level'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:2000:in `run'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:2067:in `standard_exception_handling'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/lib/rake.rb:1997:in `run'
/usr/lib/ruby/gems/1.8/gems/rake-0.8.7/bin/rake:31
/usr/bin/rake:19:in `load'
/usr/bin/rake:19
```

Poking around, I found that the problem was references to family\_id that should have been @family\_id. Simple patch attached.

## Associated revisions

---

### Revision 1b1e1820 - 06/02/2010 08:31 AM - Frank Sweetser

fixes #296 Check for existence of family\_id before referencing it

## History

---

### #1 - 06/02/2010 03:09 AM - Ohad Levy

- Category set to DB migrations
- Assignee set to Ohad Levy
- Target version set to 0.1-5

this does not make any sense to me (as @family\_id is not defined anywhere).

maybe a simple rescue statement in that method should be done (e.g. if it failed to load the family data)

### #2 - 06/02/2010 03:24 AM - Frank Sweetser

- File 0001-Check-for-existence-of-family\_id-before-referencing-.patch added
- Assignee deleted (Ohad Levy)
- Target version deleted (0.1-5)

Actually, ignore that previous patch, as digging more it's completely wrong. This one seems to work better for me.

### #3 - 06/02/2010 03:28 AM - Ohad Levy

did you had a chance to remigrate your original data?  
the defined? was part of the original patch, which I fixed a while ago with respond\_to.

I see no damage in including this patch, just based on past experience, I'm not sure if that's the root cause.

maybe add a rescue to be on the safe side? :)

Thanks,  
Ohad

### #4 - 06/02/2010 03:47 AM - Frank Sweetser

Yes, I rolled back to my pre-1.5 database snapshot and re-ran the migrations from scratch. So far, it Works For Me.

My ruby-fu is far to weak to know if the defined? is the best solution or not. If it helps, I can reproduce the problem quickly in irb:

```
[root@jms foreman]# ./script/console
Loading production environment (Rails 2.3.5)
```

```
Operatingsystem.exists?(1)
```

ActiveRecord::MissingAttributeError: missing attribute: family\_id

With the defined?, this works and returns true.

**#5 - 06/02/2010 03:54 AM - Ohad Levy**

- Status changed from New to Ready For Testing

- Assignee set to Ohad Levy

ok, I'll play a bit more and merge soon :)

**#6 - 06/02/2010 03:55 AM - Ohad Levy**

- Target version set to 0.1-5

**#7 - 06/02/2010 08:43 AM - Frank Sweetser**

- % Done changed from 0 to 100

Applied in changeset [1b1e1820c2a07cda29d9b52acfe333aa75b7f84e](#).

**#8 - 06/02/2010 02:02 PM - Ohad Levy**

- Status changed from Ready For Testing to Closed

**Files**

---

|                                                                 |           |            |                |
|-----------------------------------------------------------------|-----------|------------|----------------|
| 0001-Fix-family_id-references.patch                             | 873 Bytes | 06/02/2010 | Frank Sweetser |
| 0001-Check-for-existence-of-family_id-before-referencing-.patch | 924 Bytes | 06/02/2010 | Frank Sweetser |